

Variables & Algorithms

Introduction

To do arithmetic, or count things inside a Java program, you must use variables. To write a program, first you must come up with an algorithm.

Some facts about variables

1. Variables are containers for information. The purpose of variables is to hold information. That information may be of various types; numbers, letters, words, or even objects. Java variables are similar to, but different from, variables in algebra. Like in mathematics, Java variables might contain any number of values. The particular values of all the variables in a program at a particular moment determine the state of the computation the program is performing (*that's a mouthful, eh?*).
2. There is only one way to change the information inside a variable, which is to execute an assignment statement. In algebra, you might encounter a formula like $x = y * 2$, and be asked, if $y=2$, then what is x ? In Java, by contrast, $x = y * 2$; means take the value of y , multiply it by 2, and store the resulting value in the variable x (or assign that value to x). So, if the value of y were 17, and that assignment statement were executed, afterwards the value of x would be 34.
3. Variables hold exactly one value at a time. Whatever value had been in x before the assignment statement was executed is irretrievably lost. If you wanted to keep the old value, you would need another variable to hold it.
4. Every variable has three attributes: name, type and value. Java is what is called a typed language. In addition to a name and a value, every variable in Java has a type. For now we will only consider variables of two built-in types. The first you have already seen; `String` is used to store a series of literal characters. The second, `int`, is used to store whole numbers, like 12, 1,000,000, or -37.

Declaring variables

Variables store information; the state of an object is determined by the value of its variables. Every variable has a name, a type, and a value. When a variable is declared, only the name and type are required. Here are some examples:

```
int count; // this declares an int variable named count
String name; // this declares a String variable named name
```

Having declared (and thus created) those two variables, they can be used:

```
System.out.println("Count=" + count); // this will print count=0
count = 17; // this sets count to 17
System.out.println("Now, count=" + count); // this will print Now, count=17

name = "frank";
System.out.println(name); // this will print frank
```

You can change the value of a variable by doing arithmetic:

```
int count=0;
count = count + 1; // after this count will be 1
count = count + 1; // after this count will be 2
```

Algorithms

A program is an algorithm written in a particular programming language. That's fine, but what's an algorithm?

Rough Definition: An algorithm is a step-by-step description of a process to solve a problem.

Thus a recipe is, roughly, an algorithm. It lists the various ingredients that should be added in what order, and the cooking or baking process. Most recipes are not quite algorithms because they require judgment to carry out correctly.

Better definition: An algorithm is a step-by-step description of a process, where each step is described explicitly and requires no judgment, to solve a problem, or a class of problems.

The reason many recipes do not fit this definition is that they include directions like, "Bake until done", or "Cook until just tender.". While any experienced cook understands these instructions, the cook must exercise judgment to follow them. People do this very well, but computers do not.

So, when presented with a problem to be solved by a program, a programmer's job is first to formulate an algorithm which will solve that problem, and then to convert it into a programming language, so that it can be executed (i.e. carried out) by a computer.